AMENDMENTS TO THE CLAIMS

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Presented) In a distributed computing environment that typically uses a single distributed transaction coordinator to monitor changes to files on a per logical system volume basis, a series of computer programs with computer executable instructions within one or more computer storage medium used to provide a plurality of independent resource managers that operate with respect to transactions, thereby acting as separate units within the volume in order to ensure that operations to one database within the volume do not affect operations of other databases or users of the volume, the series of computer programs comprising:

a logical volume of a file system that includes a plurality of files or resources among a distributed system, wherein one or more of the plurality of files or resources within the file system is controlled by a particular resource manager, which is an object that participates in a transaction and provides a subsystem that implements a transaction-protected resource that monitors and controls changes to files or resources; and

a plurality of resource managers maintained on the file system volume, each resource manager independent from one another such that a file or resource monitored and controlled by a particular resources manager cannot be monitored or controlled by any other resource manager from the plurality of resource managers, and wherein each resource manager from the plurality of resources managers has associated transactional metadata and a collection of associated files, wherein the transactional metadata is maintained based on a scope of control set for each of the plurality of resource managers by defining a collection of files or resources based on one or more of a directory hierarchy, a file extension, a file type, a timestamp, a file size, or a tag within the files for which the particular resource manager is responsible in order to allow various options offering different levels of performance, reliability, feature availability, and manageability on a per-resource basis rather than a per volume basis.

2. (Previously Presented) The series of computer programs of claim 1 wherein at least one resource manager comprises properties that differ from properties of another resource manager.

3. (Previously Presented) The series of computer programs of claim 1 wherein at least one resource manager comprises transactional file system metadata that differ from transactional file system metadata of another resource manager.

4. (Previously Presented) The series of computer programs of claim 1 wherein one of the resource managers contains files associated with a first database, and wherein another of the resource managers contains files associated with a second database.

5. (Previously Presented) The series of computer programs of claim 1 wherein the file system maintains a volume control data structure associated with a set containing at least one resource manager control data structure.

6. (Previously Presented) The series of computer programs of claim 1 further comprising a mechanism in the file system for discovering a resource manager control data structure associated with a file data structure.

7. (Previously Presented) The series of computer programs of claim 1 wherein the file system maintains a first data structure having data identifying at least one resource manager control data structure.

8. (Previously Presented) The series of computer programs of claim 7 wherein each file in the collection includes a reference to data maintained in the first data structure to identify a resource manager control data structure for that file.

9. (Previously Presented) The series of computer programs of claim 1 further comprising an open file object on the volume, wherein the file system maintains a file control data structure corresponding to the open file object, the file control data structure including a reference to a

resource manager control data structure that corresponds to a resource manager to which the file is associated.

10. (Previously Presented) The series of computer programs of claim 9 wherein the file control data structure includes data that indicates that the open file object comprises the resource manager.

11. (Previously Presented) The series of computer programs of claim 9 wherein the data is persisted in a record in a file system table, the record corresponding to the file.

12. (Previously Presented) The series of computer programs of claim 1 wherein the file system includes a set of functions for interfacing with the resource manager.

13. (Previously Presented) The series of computer programs of claim 12 wherein one function creates a new resource manager.

14. (Previously Presented) The series of computer programs of claim 12 wherein one function starts a resource manager.

15. (Previously Presented) The series of computer programs of claim 1 wherein each resource manager corresponds to a directory hierarchy, and wherein the collection of associated files comprises files logically under that directory hierarchy.

16. (Previously Presented) The series of computer programs of claim 1 wherein associated transactional metadata includes a log file.

17. (Previously Presented) In a distributed computing environment that typically uses a single distributed transaction coordinator to monitor changes to files on a per logical system volume basis, a method of associating a file object with a resource manager in a system that includes a plurality of independent resource managers that operate with respect to transactions, thereby acting as separate units within the volume in order to ensure that operations to one database within the volume do not affect operations of other databases or users of the volume, the method comprising:

separating a volume into a plurality of resource managers that are independent from one another such that a file or resource monitored and controlled by a particular resource manager cannot be monitored or controlled by any other resource manager from the plurality of resources managers and each resource manager associated with transaction metadata, which is maintained based on a scope of control set for each of the plurality of resource mangers by defining a collection of files or resources based on a directory hierarchy for which the particular resource manager is responsible in order to allow various options offering different levels of performance, reliability, feature availability, and manageability on a per-resource basis rather than a per volume basis;

receiving a request to open a file system object associated with a resource manager;

creating a file control block for the file system object;

determining that the file control block does not reference a resource manager control block; and

based on the determination, discovering a resource manager control block corresponding to the file system object and associating the file control block with the discovered resource manager control block by storing a pointer in the file control block that identifies the discovered resource manager control block, the resource manager responsible for the file system object, or both.

18. (Original) The method of claim 17 wherein discovering the resource manager control block includes creating a resource manager control block.

19. (Original) The method of claim 17 wherein associating the file control block with the discovered resource manager control block comprises writing a pointer into the file control block that points to the resource manager control block.

20. (Original) The method of claim 17 wherein discovering the resource manager control block includes determining whether the resource manager control block exists, and if not, creating the resource manager control block, and modifying the file control block to include an association with the resource manager control block.

21. (Original) The method of claim 17 wherein discovering the resource manager control block includes locating a file control block of a parent file that is associated with the resource manager control block.

22. (Original) The method of claim 17 wherein discovering the resource manager control block includes locating a reference to a table location containing resource manager control block data, and using the reference to obtain a pointer to the resource manager control block.

23. (Original) The method of claim 22 wherein locating the reference to the table location comprises reading a header of the file object.

24. (Original) The method of claim 22 further comprising maintaining a table including the table location in a volume control block.

25. (Original) The method of claim 17 wherein each resource manager corresponds to a subdirectory in the file system, and wherein the file system object is logically associated with the subdirectory.

26. (Original) The method of claim 17 wherein at least one resource manager is associated with a database, and further comprising, performing a transaction that includes at least one operation by the database and at least one operation by the file system.

27. (Previously Presented) A computer-storage medium having computer-executable instructions for performing the method of claim 17.

45. (Currently Amended) In a distributed computing environment that typically uses a single distributed transaction coordinator to monitor changes to files on a per logical system volume basis, a method ~~or~~ of providing a plurality of independent resources managers that operate with respect to transactions, thereby acting as separate units within the volume in order to ensure that operations to one database within the volume do not affect operations of other databases or users of the volume, the method comprising:

separating a file system volume into a plurality of transactional resource managers that provide transactional services, wherein each of the transactional resource managers maintain transactional metadata based on a scope of control set for each of the plurality of resource mangers by defining a collection of files or resources for a particular transactional resource manager based on one or more of a directory hierarchy, a file extension, a file type, a timestamp, a file size, or a tag within the files for which the particular transactional resource manager is responsible in order to allow various options offering different levels of performance, reliability, feature availability, and manageability on a per-resource basis rather than a per volume basis; and

based on the scope of control set for a selected resource manager, monitoring of a transaction and controlling access to one or more files or resources with respect to the selected resource manager, the resource managers being independent of one another such that the monitoring of the transaction and controlling of the one or more files or resources cannot be performed by any other transactional resource manager within the plurality of transactional resource managers.

46. (Original) The method of claim 45 further comprising, receiving a request to perform the function.

47. (Original) The method of claim 46 wherein receiving the request comprises receiving an application programming interface call.

48. (Original) The method of claim 45 wherein the function corresponds to a backup operation of at least some of the files of a resource manager.

49. (Original) The method of claim 45 wherein the function corresponds to a restore operation of at least some of the files of a resource manager.

50. (Original) The method of claim 45 wherein the function corresponds to a roll forward to a point in time operation.

51. (Original) The method of claim 45 wherein the function corresponds to a crash recovery operation.

52. (Original) The method of claim 45 wherein the function corresponds to a redo phase of a recovery operation.

53. (Original) The method of claim 52 further comprising, performing the function at least one other time.

54. (Original) The method of claim 45 wherein the function corresponds to an undo phase of a recovery operation.

55. (Previously Presented) A computer storage medium having computer-executable instructions for performing the method of claim 45.

56. (New) In a distributed computing environment, a method of determining a resource manager associated with a file in order to allow the resource manager to monitor transactional relations for that file within a volume, the method comprising:

identifying an opening of a file object chosen from a plurality of file objects within a volume of a file system, wherein transactions associated with the plurality of file objects are monitored by a corresponding resource manager chosen from a plurality of resource managers for the volume, and wherein each of the plurality of resources managers is a file object that participates in a file object transaction and provides a subsystem that implements a transaction-protected resource that monitors and controls changes to file objects within its scope defined by the collection of file objects for which it is responsible such that a file object within a resource managers scope is associated only with that resource manager and the resource manager independently maintains a set of metadata and provides transactional capabilities for all file objects within its scope, which guarantees transactional consistency;

based on the opening of the file object, creating a file control block that provides an in-memory data structure in which the file system maintains state for the open file object, wherein the file control block forms part of a hierarchical data structure within the file system's directory tree, which includes a root file object representing a default resource manager for files not managed by other resource managers in the file system and at least one subdirectory file object under the root file object representing another resource manager within the directory tree, wherein both the root file object and the at least one subdirectory file object include special properties and attributes, including per-resource manager transactional-related metadata maintained on a per resource manager basis, rather than on a per volume basis, and wherein the default resource manage becomes the default depository for transactional metadata for all files that are not part of the at least one subdirectory resource manager;

determining if the file control block includes a valid pointer to a resource manager control block, wherein if no valid control block exists for the resource manager control block, the method further comprises;

determining if the file control block is flagged as the at least one subdirectory file object in the directory tree, wherein if the control block is flagged as such, a new resource manager control block for the file object is created and a pointer to the new resource manager control block is stored in a field for the file control block created, and wherein if the file control block is not flagged as the subdirectory file object, the method further comprises:

determining whether a file control block exists for a parent file object in the directory tree, wherein if the file control block for the parent exists, determining a resource manager control block for the parent file object and assigning a pointer to the file control block to the file object opened that references the resource manager control block for the parent file object, and wherein if no parent file control block exists, the method further comprises:

traversing the directory tree until a higher parent with a valid resource manager control block is found or a higher patent is flagged as the root file object or the at least one subdirectory file object, wherein if a valid resource manager control block is found for the higher parent, a pointer to the valid resource manager control block is including in a field of at least the file control block for the file object open, and wherein if the control block for the higher parent is flagged, a new resource manager control block for the control block for the higher parent is created and a pointer thereto is included in a field for at least the file control block for the file object created.

57. (New) The method of claim 56, wherein the method further comprises:

upon creating the file control block, copying thereto a portion of data from a master file table that includes records for each of the plurality of file objects in the file system, and wherein at least one record in the master file table includes a flag attribute set indicating that the at least one subdirectory file object corresponds to a resource manager descriptor.

58. (New) The method of claim 57, wherein if a record used to copy the portion of data to the file control clock includes the flag attribute indicating that a new resource manager control block should be created, the method further comprises:

accessing a volume control block that includes a transactional file system component, which updates references to resource manager control blocks, the volume control block providing an in-memory representation of state for a volume's data used by a file system to manage the volume; and

adding information about the new resource manager control block to the transactional file system component referencing the new resource manager control block created, and wherein the transaction file system component includes a list of references to a plurality of resource manager control blocks in the volume.

59. (New) The method of claim 58, wherein the list of references to the plurality of resource manager control blocks included in the transactional file system are only for resource managers that are open such that the plurality of resource manager within the list are actively providing one or more of: transactional services, communication with a transaction manager, running recovery, or in a transition state.

60. (New) The method of claim 59, wherein a resource manager is considered closed if it simply exists in a state where it may be opened at a later time.